REMARKS

This Amendment and Response is filed in reply to the Office action dated October 31, 2007. Claims 1, 3, 19, 46, 59 and 65 are amended and claims 2, 4, 32-36, 39-45, 48, 55-58 and 67 were previously canceled. Accordingly, after entry of this Amendment and Response, claims 1, 3, 5-31, 37-38, 46-47, 49-54, 59-66 and 68-69 remain pending.

I. Specification

The specification is objected to because it contains an embedded hyperlink and/or other form of browser-executable code. In response, the specification has been amended to remove the embedded hyperlinks.

II. Claim Rejections Under 35 U.S.C. § 112

Claims 1, 3, 5-18, 19-31, 37-38, 46-47, 49, 60, 62-66 and 68-69 are rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement. First, independent claims 1, 19 and 59, as amended in the last Office action, recite that the claimed backtracking is performed strictly within an unambiguous skid region while dependent claims 3, 31, 60 and 63-69 recite the limitation of an ambiguity location. The Examiner believes that the specification does not have any disclosure describing an unambiguous skid region containing an ambiguity creating location. In response, independent claims 1, 19 and 59 are amended to clarify that the backtracking is not performed strictly within an unambiguous skid region. Claims 1, 19 and 59, as currently amended, recite the limitation "determining whether the ambiguity creating location is encountered during backtracking." That is, an ambiguity creating location may or may not be encountered during backtracking. As such, the backtracking is not restricted to being performed in an unambiguous skid region.

Second, claim 46 recites the limitation of "classifying the execution event as associated with one or more of the ambiguity creating locations or not associated with one or more of the ambiguity creating locations." The Examiner believes that the specification does not have any disclosure to support this limitation. While we do not necessarily agree, claim 46 is amended to recite "associating the execution event with one or more of the ambiguity creating locations when one or more of the ambiguity creating locations are encountered while backtracking." Support for this amendment may be found at least at paragraph 1038 of the specification.

Claims 1, 3, 5-18, 19-31, 37-38, 60, 62-66 and 68 are rejected under U.S.C. § 112, second paragraph, as failing to set forth the subject matter which applicant(s) regard as their invention. Specifically, claims 1, 19 and 59 recite "determining that the ambiguity creating

location is not encountered while backtracking." The Examiner asserts that the purpose of the present invention is to perform backtracking while in a skid region containing ambiguity creating locations (relying on a statement in the specification that absent the ambiguity creating location, backtracking from the miss detection to the load instruction is straightforward). Claims 1, 19 and 46, as currently amended, recite "determining whether the ambiguity creating location is encountered during backtracking." As such, the claims contemplate backtracking through a skid region containing an ambiguity creating location or a skid region that does not contain an ambiguity creating location. As discussed in more detail in section III, when latency exists between the operation causing the event and detection of the event, backtracking to associate the event with the operation is complicated by ambiguity creating locations and the type of operation causing the event (which may limit how far back to go depending on the number of instructions which can be executed before the event occurs). As such, the Applicant does not regard the invention as being limited to performing backtracking only in skid regions containing an ambiguity creating location. For example, the specification states that as backtracking progresses, if a target instruction is encountered without an intervening ambiguity creating location, then the detected execution event is associated with the target instruction. On the other hand, if an intervening ambiguity creating location is encountered, the execution event may be discarded or the ambiguity creating location may be bridged using a branch history queue. *See Specification, paragraph 38.*

The Applicant respectfully submits that the claims, as amended, comply with 35 U.S.C. § 112, first and second paragraphs, and respectfully requests such indication.

III. Claim Rejections Under 35 U.S.C. § 102

Claims 1, 3, 5-15, 19-31, 37-38, 46-47, 49, 59-60 and 62 are rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,964,867 to Anderson et al. (hereinafter "Anderson"). A proper anticipation rejection requires that each and every limitation of a claim be disclosed in a single prior art reference.

Initially, the rejections of independent claims 1, 19, 46 and 59 are addressed. The invention set forth in the independent claims is directed to the problem of profiling code when latency exists between an execution event and the detection of the execution event. When latency exists, the program counter returned at the time of detection (the detection point) represents the next instruction to issue rather than the instruction (or operation) causing the execution event (this may be referred to as program counter skid). Backtracking through the code is performed to associate the execution event with the operation causing the event. Backtracking may be complicated by the presence of an ambiguity creating location (i.e., an instruction reachable from more than one instruction) between the operation causing the

execution event and the current instruction at which the event is detected. A code preparation facility (compiler or assembler/linker) may store the program counter value of every ambiguity creating location (e.g., the program counter value of a branch target). A check may be done to determine if a stored program counter value is encountered during backtracking, indicating that an ambiguity creating location was encountered. Further, how far to backtrack depends on the type of instruction causing the execution event as well as the processor architecture. For example, detection of a cache miss on an in-order, single instruction issue per cycle processor may occur six instructions later. However, detection of a cache miss on an out-of-order processor capable of issuing, four instructions per cycle may occur up to 24 instructions later.

A. Anderson Does Not Disclose Backtracking A Displacement From A Detection Point Based On The Type Of Operation Triggering The Execution Event

More specifically, independent claim 1 includes the limitations "backtracking a displacement from a detection point in the code coinciding with the detection of the execution event to a preceding operation" and "wherein the displacement is based, at least in part, on a type of operation appropriate to have triggered the execution event." Independent claims 19, 46 and 59 include similar limitations. That is, the independent claims require that the displacement from the detection point of the event used for backtracking be based on the type of operation causing the event. The Office action alleges Anderson discloses these limitations. *See Office action, page 5-6, citing Anderson, column 3, lines 35-49.* The Applicant respectfully disagrees for at least the following reasons.

Anderson discloses that the main problem with event counters is that the amount of the delay is an unpredictable amount. *See Anderson, column 3, lines 22-27.* Anderson further discloses that static analysis can sometimes work backwards to identify the instruction causing the event by attributing all performance counter events to the instruction that is executing six cycles after the event. *See Anderson, column 3, lines 40-49.* As such, Anderson always uses a fixed displacement of six instructions to associate an event with the instruction causing the event. Anderson does not disclose using a displacement based on a type of operation causing the event to associate the event with the instruction causing the event, as required by the independent claims. For at least this reason, Anderson does not anticipate independent claims 1, 19, 46 and 59.

B. Anderson Does Not Disclose Identifying Ambiguity Creating Locations During Machine Code Generation

Independent claim 1, as amended, further requires "during machine code generation, identifying an ambiguity creating location in the code." Support for this amendment may be found at least at paragraph 1024 and Figure 3, operations 301, 302 and 312 of the specification. Independent claims 19, 46 and 59 each include a similar limitation. The Office

action cites Anderson, column 23, line 55 to column 24, line 9 as teaching this limitation. *See Office action, page 5.* The Applicant respectfully disagrees for at least the following reasons.

Anderson discloses backward analysis of a control flow graph of the program. *See Anderson, column 24, 26-27.* Sometimes branch history bits allow identification of an execution path consistent with the values of the branch history bits when there is a merge in the control flow graph. *See Anderson, column 24, lines 36-38.* That is, Anderson teaches the use of a control flow graph and branch history bits to identify an execution path taken. Anderson does not disclose identifying ambiguity creating locations during machine code generation as required by the independent claims. For at least this reason, Anderson does not anticipate independent claims 1, 19, 46 and 59.

C. Conclusion

For at least the above stated reasons, Anderson does not disclose all of the limitations of independent claims 1, 19, 46 and 59. Insofar as Anderson does not disclose all of the limitations of independent claims 1, 19, 46 and 59, Anderson is inadequate to anticipate independent claims 1, 19, 46 and 59 and such indication is respectfully requested.

The remaining rejected claims 3, 5-15, 20-31, 37-38, 47, 49, 60 and 62 all depend, either directly or indirectly, from one of independent claims 1, 19, 46 and 59. Accordingly, these dependent claims are themselves patentable over Anderson under 35 U.S.C. 102(b) for at least the same reasons and such indication is respectfully requested. This statement is made without reference to or waiving the independent bases of patentability within each dependent claim.

IV. Claim Rejections Under 35 U.S.C. § 103

Claims 16-18 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Anderson in view of U.S. Patent Application Publication No. 2002/0010913 to Ronstrom (hereinafter "Ronstrom"). Claims 63-66 and 68-69 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Anderson in view of "Efficient Instruction Scheduling Using Finite State Automata," *Proceedings of MICRO-28*, Vasanth Bala and Norman Rubin (hereinafter "Bala"). Claims 50-54 and 61 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Ronstrom in view of Bala. A proper prima facie case of obviousness requires that the cited references teach or suggest all of the claim limitations. *See MPEP § 2143.*

A. Rejection of claims 16-18, 63-66 and 68-69

Claims 16-18, 63-66 and 68-69 depend, either directly or indirectly, from one of independent claims 1, 19 and 46. The Office action relies on Anderson to disclose the limitations "backtracking a displacement from a detection point in the code coinciding with the detection of the execution event to a preceding operation," "wherein the displacement is

based, at least in part, on a type of operation appropriate to have triggered the execution event" and "during machine code generation, identifying an ambiguity creating location in the code." However, for the reasons set forth above in section III, Anderson does not disclose using a displacement based on a type of operation causing the event to associate the event with the instruction causing the event nor identifying ambiguity creating locations during machine code generation. Further, neither Ronstrom nor Bala are sufficient to remedy the deficiency of Anderson. Accordingly, these dependent claims are patentable over Anderson and Ronstrom as well as Anderson and Bala because the combined references do not teach or suggest all of the limitations of the independent claims 1, 19 and 46 from which dependent claims 16-18, 63-66 and 68-69 depend. This statement is made without reference to or waiving the independent bases of patentability within the dependent claims. The applicant, therefore, respectfully requests withdrawal of this rejection.

B. Rejection of claims 50-54 and 61

Claims 50-54 and 61 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Ronstrom in view of Bala. Initially, the rejections of independent claims 50 and 61 are addressed.

Claim 61 recites a limitation "padding the execution sequence to provide an unambiguous skid region between a particular operation and a subsequent ambiguity creating location within the sequence of operations to cover an expected detection latency." Claim 50 recites a similar limitation. The Office action relies on Bala to teach the above recited limitations of claims 50 and 61. *See Office action, pages 22-23, citing Bala, page 46, section 1, lines 6-8 and page 47, section 1, last paragraph lines 6-10.* The Applicant respectfully disagrees for at least the following reasons.

Bala teaches techniques for avoiding pipeline hazards when an instruction is speculatively executed above a branch upon which it is control dependent. *See Bala, page 46, section 1, paragraph 1, lines 1-8 and page 47, section 1, last paragraph.* That is, Bala discloses techniques for relocating instructions above a branch that are not dependent upon the branch being taken or not taken such that the reordering of instructions preserve control dependences. In contrast, the independent claims require padding the execution sequence to provide an unambiguous skid region between a particular operation and a subsequent ambiguity creation location within the sequence of operations to cover an expected detection latency. As such, Bala does not disclose or suggest "padding the execution sequence to provide an unambiguous skid region between a particular operation and a subsequent ambiguity creating location within the sequence of operations to cover an expected detection latency" as required by independent claims 50 and 61. Further, Ronstrom is inadequate to remedy the deficiency of Bala.

Accordingly, the Applicant respectfully submits that independent claims 50 and 61 are patentable over Ronstrom in view of Bala and such indication is respectfully requested. The remaining rejected claims 51-54 depend from, either directly or indirectly, from independent claim 50. Accordingly, these dependent claims are themselves patentable over Ronstrom in view of Bala and such indication is respectfully requested. This statement is made without reference to or waiving the independent bases of patentability within each dependent claim.

V. Conclusion

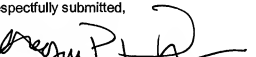
The Applicant thanks the Examiner for his thorough review of the application. The Applicant respectfully submits the present application, as amended, is in condition for allowance and respectfully requests the issuance of a Notice of Allowability as soon as practicable.

This Amendment is submitted contemporaneously with a petition for a one-month extension of time in accordance with 37 C.F.R. § 1.136(a). Accordingly, please charge Deposit Account No. 04-1415 in the amount of $120.00, for a one-month extension of time fee. The Applicant believes no further fees or petitions are required. However, if any such petitions or fees are necessary, please consider this a request therefor and authorization to charge Deposit Account No. 04-1415 accordingly.

If the Examiner should require any additional information or amendment, please contact the undersigned attorney.

Dated: Feb. 29, 2008

Respectfully submitted,



Gregory P. Durbin, Registration No. 42,503
Attorney for Applicant
USPTO Customer No. 66083

DORSEY & WHITNEY LLP
Republic Plaza Building, Suite 4700
370 Seventeenth Street
Denver, Colorado 80202-5647
Phone: (303) 629-3400
Fax: (303) 629-3450